

PATENT ABSTRACTS OF JAPAN

(11)Publication number : 10-239394

(43)Date of publication of application : 11.09.1998

(51)Int.Cl.

G01R 31/28
G06F 17/50

(21)Application number : 09-039122

(71)Applicant : NEC CORP

(22)Date of filing : 24.02.1997

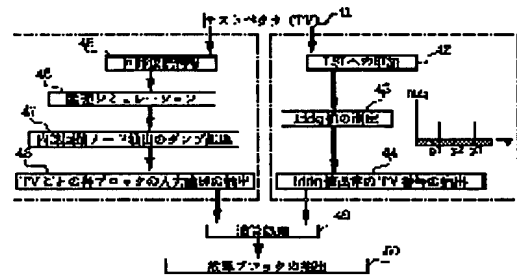
(72)Inventor : SANADA KATSU

(54) METHOD FOR DETERMINING DEFECTIVE PORTION OF LSI

(57)Abstract:

PROBLEM TO BE SOLVED: To provide a method for determining a defective portion of an LSI capable of readily and quickly narrowing the multiple defective portions in the LSI and determining the number of defectives and the defective portions by small data amount irrespective of malfunction of an output terminal.

SOLUTION: Logical information 48 of each block that changes by each test vector 41 is outputted by a dump operation 47 from circuit connection information 45 by using logical simulation 46 for inspection of an LSI circuit. The logical information 48 of each block by each test vector 41 is combined with information 44 of presence or absence of Iddq malfunction (leakage current in a static condition) so that an arithmetic operation 49 of each block by each test vector is executed and extraction 50 of a block including a defective portion is executed. The defective portion at a transistor level is extracted from the logical information in terms of the presence or absence of Iddq malfunction at the block.



LEGAL STATUS

[Date of request for examination] 24.02.1997

[Date of sending the examiner's decision of rejection]

[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]

[Date of final disposal for application]

[Patent number] 2947204

[Date of registration] 02.07.1999

[Number of appeal against examiner's decision of rejection]

[Date of requesting appeal against examiner's decision of rejection]

[Date of extinction of right]

(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号

特開平10-239394

(43) 公開日 平成10年(1998) 9月11日

(51) Int.Cl.⁶

識別記号

F I

G 0 1 R 31/28

G 0 1 R 31/28

F

G 0 6 F 17/50

G 0 6 F 15/60

6 7 2 D

審査請求 有 請求項の数9 OL (全 15 頁)

(21) 出願番号 特願平9-39122

(22) 出願日 平成9年(1997) 2月24日

(71) 出願人 000004237

日本電気株式会社

東京都港区芝五丁目7番1号

(72) 発明者 真田 克

東京都港区芝五丁目7番1号 日本電気株式会社内

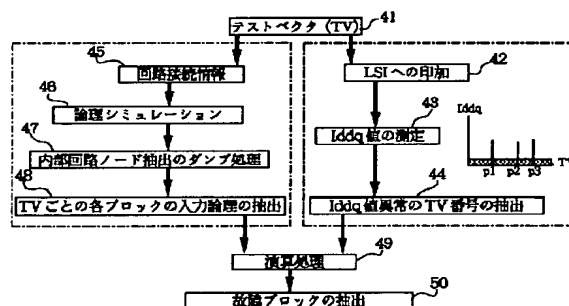
(74) 代理人 弁理士 若林 忠

(54) 【発明の名称】 L S I の故障箇所の特定化方法

(57) 【要約】

【課題】 L S I 内部の多重故障を、少ないデータ量で、出力端子異常に無関係に、容易にかつ高速で絞り込み、故障数および故障箇所を特定化できる L S I の故障箇所の特定化方法を提供する。

【解決手段】 テストベクタ41ごとに変化する各ブロックの論理情報48は、回路接続情報45から L S I 回路検証のための論理シミュレーション46を用いてダンブ処理47によって出力される。そのテストベクタごとの各ブロックの論理情報48をテストベクタごとの I d d q 異常の有無の情報44と組み合わせることで、各ブロックのテストベクタごとの演算処理49が実施され、故障箇所を内蔵したブロックの抽出50が行なわれ、そのブロックでの I d d q 異常の有無における論理情報からトランジスタレベルでの異常箇所が抽出される。



【特許請求の範囲】

【請求項1】 LSIの入力端子より順次入力されるテストベクタに従って変化する、前記LSIを構成する基本的論理回路単位であるブロックごとの論理動作情報と、該テストベクタにより生成する、前記LSIの論理動作の静止状態でのリーク電流であるIddqの値が所定値を越えるテストベクタ番号とを用いて、該ブロックごとの論理演算を行うことにより故障ブロックを抽出し、次に該故障ブロックにおけるIddq異常を有する前記テストベクタ番号での論理情報を用いて多重故障箇所を絞り込む、ことを特徴とするLSIの故障箇所の特定化方法。

【請求項2】 ブロックごとの前記論理動作情報は、LSIの入力端子より入力する前記テストベクタに同期して変化する各ブロックごとの前記テストベクタ番号ごとの入力論理の組合わせである、請求項1に記載のLSIの故障箇所の特定化方法。

【請求項3】 ブロックごとの論理演算を行うことによる前記故障ブロック抽出は、Iddq異常が発生する複数種類の入力論理の組合わせの全てが、Iddq値が正常なテストベクタ番号での入力論理の組合わせに存在するときのみ、該ブロックを正常ブロックとして抽出し、それ以外は該ブロックを故障ブロックとして抽出する、請求項1に記載のLSIの故障箇所の特定化方法。

【請求項4】 ブロックごとの論理演算を行うことによる前記故障ブロック抽出は、Iddq異常が連続したテストベクタ番号にて検出される複数種類の連続した入力論理の組合わせ群の全てが、Iddq値が正常な連続したテストベクタ番号での入力論理の組合わせ群に存在するときのみ、該ブロックを、正常ブロックとして抽出し、それ以外は該ブロックを故障ブロックとして抽出する、請求項1に記載のLSIの故障箇所の特定化方法。

【請求項5】 ブロックごとの論理演算を行うことによる前記故障ブロック抽出は、Iddq異常が連続したテストベクタ番号にて検出される複数種類の入力論理の組合わせ群、およびIddq異常が連続しないテストベクタ番号にて検出される複数種類の入力論理の組合わせの全てが、Iddq値が正常な連続したテストベクタ番号での入力論理の組合わせ群、およびIddq値が正常なテストベクタ番号での入力論理の組合わせに存在するときのみ、該ブロックを正常ブロックとして抽出し、それ以外は該ブロックを故障ブロックとして抽出する、請求項1に記載のLSIの故障箇所の特定化方法。

【請求項6】 前記故障ブロックにおける前記故障箇所の絞り込みは、該故障ブロックにおけるIddq異常が発生している前記テストベクタ番号での入力論理の組合わせで故障箇所を絞り込む、請求項1に記載のLSIの故障箇所の特定化方法。

【請求項7】 前記故障ブロックにおける前記故障箇所の絞り込みは、該故障ブロックにおけるIddq異常が

発生している前記テストベクタ番号での入力論理の組合わせとIddq異常が発生していないテストベクタ番号での入力論理の組合わせとを用いて故障箇所を絞り込む、請求項1に記載のLSIの故障箇所の特定化方法。

【請求項8】 前記ブロックごとの論理演算を行うことによる故障ブロック抽出は、組合わせ回路と順序回路とを分離して故障箇所を絞り込む、請求項1に記載のLSIの故障箇所の特定化方法。

【請求項9】 前記ブロックごとの論理演算を行うことによる故障ブロック抽出は、回路構成を複数の基本的論理回路から順に回路規模を縮小していくことにより故障箇所を絞り込む、請求項1に記載のLSIの故障箇所の特定化方法。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】本発明はCMOSLSIの故障箇所絞り込みに関し、特に静止状態電源電流の発生有無とテストベクタにおける内部回路の論理状態とを比較することにより多重故障箇所を特定する、CMOS論理回路の故障箇所絞り込み方法に関する。

【0002】

【従来の技術】従来のCADを利用したシミュレーションによりLSIの故障箇所を絞り込む方法は、出力端子に異常が発生した情報をもととして推定する方法であった。

【0003】その第一の方法は故障辞書の作成による故障シミュレーション方法であり、この方法は内部回路の各ブロックに故障を定義しながら、異常が発覚する出力端子、出力値、そしてテストパターン番号を、実際の故障品のデータと比較することで故障箇所を推定する方法であった。

【0004】図12は従来例の故障辞書作成による故障シミュレーション方法の流れブロック図であり、図中符号121はテストベクタ、122は故障定義、123、124はLSIの論理接続、125、126は論理シミュレーション、127は比較である。故障箇所の指定には具体的には図12に示すように、LSIの出力端子における論理シミュレーション126による正常論理値と、内部回路の各ノードに故障定義122を行なった論理シミュレーション125による故障論理値とを用い、LSIの入力端子にテストベクタ121を入力した時、出力端子より出力する論理の比較127により実際の故障品と一致する故障定義位置を抽出する方法である。

【0005】第二の方法はバックトレース方法と称し、異常が発覚した出力端子、出力値、そしてテストパターン番号をもとに、出力端子から入力端子方向へ論理を逆にトレースする方法である。すなわち、LSIの入力端子に所定の信号を入力したとき、出力端子に出力する信号が期待値と異なっていたとき、その出力値と期待値との相違を利用して、出力端子から入力端子へ向かって内

部に拡散していく信号中から故障を伝搬している信号を抽出し、故障箇所を推定し、その箇所に故障を定義して再度、論理シミュレーションを行うことにより実際の故障との一致を検証する方法である。通常、複数の出力異常箇所を調査し、それらの組み合わせにより疑似故障信号を限定しながら故障箇所を絞り込んでいくのが一般的であった。

【0006】

【発明が解決しようとする課題】上述した従来の方法は、いずれも多重故障品の解析が不可能であるという点、およびシミュレーションデータが膨大となることから実用的ではなかった。

【0007】まず、多重故障品の解析に関しては、多重故障数が分からねば解析できないという大きな問題点があった。すなわち、上述した各従来方式は出力端子の異常データのみでは何個の故障が存在するか不明であるため、解析者は出力端子の異常データから故障数を推定して解析を行っていた。しかしながら、故障数の推定が間違っていた時、間違った個数に対してシミュレーションを行うため、検出される結果は完全に間違った結果に終わっていた。

【0008】次に、従来方法の各々について課題を説明する。まず、第一の方法である故障辞書作成による故障シミュレーション方法は、扱える故障モデルは単一縮退故障（Stuck-at-0、Stuck-at-1）のみであり、オープン故障はシミュレーションできないため、故障モードの特定化という点からは一般的ではなかった。なぜならば、故障シミュレーションにて扱う故障はモデル化された論理故障のためであり、オープン故障は論理が定まらないからである。さらに定義する故障数は回路を構成するすべての信号線に対して順次定義していかなければならないため、膨大なデータ量となり実用的ではなかった。すなわち、定義する故障数（V0）はLSIを構成する回路素子数（L）の3乗から4乗に比例すると言われている。

【0009】 $\ln(V0) \propto (3 \sim 4) \cdot \ln(L)$

第二の方法であるバックトレース方法は出力端子異常の情報のみをデータとして使用するため、回路内部にいくつかの故障が発生しているかを判断することができなかった。さらに、バックトレースの特徴は複数個の出力異常端子が存在してはじめて故障発生箇所が限定されていくが、その前提はいくまで単一故障の場合のみであり、また、出力端子から内部回路へ方向は信号が拡大していく方向であるため、膨大な疑似故障が検出されることになり絞り込みは不可能となる欠点があった。従って、純粹にバックトレース方式のみで故障箇所を絞り込むことが困難となってきており、例えばLSIテストングシンポジウムにて報告されているように、最近ではEBT（Electron Beam Tester）のような物理解析方法とリンクさせた、非接触による電位コン

トラスト像や論理動作波形の取得により、疑似故障箇所を故障候補から消していく方法がとられるような傾向に変わってきている。

【0010】本発明の目的は、LSI内部の多重故障を、少ないデータ量で、出力端子異常に無関係に、容易にかつ高速で絞り込み、故障数および故障箇所を特定化できるLSIの故障箇所の特定化方法を提供することにある。

【0011】

【課題を解決するための手段】本発明のLSIの故障箇所の特定化方法は、LSIの入力端子より順次入力されるテストベクタに従って変化する、LSIを構成する基本的論理回路単位であるブロックごとの論理動作情報と、該テストベクタにより生成する、LSIの論理動作の静止状態でのリーク電流であるIddqの値が所定値を越えるテストベクタ番号とを用いて、該ブロックごとの論理演算を行うことにより故障ブロックを抽出し、次に該故障ブロックにおけるIddq異常を有するテストベクタ番号での論理情報を用いて多重故障箇所を絞り込む。

【0012】ブロックごとの論理動作情報は、LSIの入力端子より入力するテストベクタに同期して変化する各ブロックごとのテストベクタ番号ごとの入力論理の組合わせであることが望ましい。

【0013】ブロックごとの論理演算を行うことによる故障ブロック抽出は、Iddq異常が発生する複数種類の入力論理の組合わせの全てが、Iddq値が正常なテストベクタ番号での入力論理の組合わせに存在するときのみ、該ブロックを正常ブロックとして抽出し、それ以外は該ブロックを故障ブロックとして抽出してもよく、Iddq異常が連続したテストベクタ番号にて検出される複数種類の連続した入力論理の組合わせ群の全てが、Iddq値が正常な連続したテストベクタ番号での入力論理の組合わせ群に存在するときのみ、該ブロックを、正常ブロックとして抽出し、それ以外は該ブロックを故障ブロックとして抽出してもよく、Iddq異常が連続したテストベクタ番号にて検出される複数種類の入力論理の組合わせ群、およびIddq異常が連続しないテストベクタ番号にて検出される複数種類の入力論理の組合わせの全てが、Iddq値が正常な連続したテストベクタ番号での入力論理の組合わせ群、およびIddq値が正常なテストベクタ番号での入力論理の組合わせに存在するときのみ、該ブロックを正常ブロックとして抽出し、それ以外は該ブロックを故障ブロックとして抽出してもよい。

【0014】故障ブロックにおける故障箇所の絞り込みは、該故障ブロックにおけるIddq異常が発生しているテストベクタ番号での入力論理の組合わせで故障箇所を絞り込んでもよく、該故障ブロックにおけるIddq異常が発生しているテストベクタ番号での入力論理の組

合わせとIddq異常が発生していないテストベクタ番号での入力論理の組み合わせとを用いて故障箇所を絞り込んでもよい。

【0015】ブロックごとの論理演算を行うことによる故障ブロック抽出は、組み合わせ回路と順序回路とを分離して故障箇所を絞り込んでもよく、回路構成を複数の基本的論理回路から順に回路規模を縮小していくことにより故障箇所を絞り込んでもよい。

【0016】CMOS論理回路は回路内部に物理欠陥を有すると、一般的傾向として“Iddq (Quiescent Vdd Supply Current)”と称する静止状態電源電流に異常値が現れる。従って、このIddq異常は、LSI回路内部の物理故障を顕在化させるシグナルとみなすことができる。この詳細については、M. Sanada “Evaluation and Detection of CMOS-LSI with Abnormal Iddq” Microelectronics and Reliability, Vol. 35, NO. 3, pp. 619-629, 1995にて明かであり、本発明は上述した性質を利用したものである。

【0017】通常、ゲートアレイ品に代表されるASIC (Application Specific Integrated Circuits) は予め準備された“ブロック”と称する基本的な論理を構成する回路を組み合わせることにより所望の電気回路を構成することにより実現される。

【0018】本発明による多重故障箇所の絞り込み方法は、上述した設計方式を利用するものであり、LSIの入力端子より入力するテストベクタに従って変化する、“ブロック”と称するLSIを構成する基本的論理回路単位での論理動作情報と、そのテストベクタごとに“Iddq”と称するLSIの論理動作の静止状態でのリーク電流の値が所定値を越えるテストベクタ番号とを用いて、ブロックごとの論理演算を行うことにより故障ブロック抽出し、次に該故障ブロックにおけるIddq異常を有するテストベクタ番号での論理情報を用いて故障箇所を絞り込む多重故障箇所の特定化方法であり、上述したブロックごとの論理動作情報はLSIの入力端子より入力するテストベクタに同期して変化する各ブロックごとのテストベクタ番号ごとの入力論理の組み合わせである。

【0019】上述のデータを用いたブロックごとの論理演算を行うことによる故障ブロック抽出方法は2種類あり、1つは、Iddq異常が発生する複数種類の入力論理の組み合わせの内の少なくとも1種類以上と同一な入力論理の組み合わせが、Iddq値が正常なテストベクタ番号における入力論理の組み合わせに存在しないブロックを故障ブロックとして抽出する方法であり、もう1つは、Iddq異常が連続したテストベクタ番号にて検出され

る複数種類の連続した入力論理の組み合わせ群の内の1種類以上と同一の入力論理の組み合わせ群が、Iddq値が正常な連続したテストベクタ番号での入力論理の組み合わせ群に存在しないブロックを故障ブロックとして抽出する多重故障箇所の特定化方法である。

【0020】さらに、上述した2種類のモード“Iddq異常が連続したテストベクタ番号にて検出される複数種類の連続した入力論理の組み合わせ群”および、“連続しないテストベクタ番号にて検出される複数種類の入力論理の組み合わせ”が混在した論理の組み合わせ中の1種類以上と同一な“入力論理の組み合わせ群”と“入力論理の組み合わせ”が、Iddq値が正常なテストベクタ番号での“入力論理の組み合わせ群”と“入力論理の組み合わせ”に存在しないブロックを故障ブロックとして抽出することを特徴とした多重故障箇所の特定化方法である。

【0021】次に、特定化された故障ブロック内部の故障箇所の絞り込みは2種類あり、一つは、その故障ブロックにおけるIddq異常が発生しているテストベクタ番号での入力論理の組み合わせを用いて故障箇所を絞り込むことを特徴としており、もう一つは、その故障ブロックにおけるIddq異常が発生しているテストベクタ番号での入力論理の組み合わせとIddq異常が発生していないテストベクタ番号での入力論理の組み合わせを用いて故障箇所を絞り込むことを特徴としている。

【0022】上述のデータを用いたブロックごとの論理演算を行うことによる故障ブロック抽出は、組み合わせ回路と順序回路を分離したことを特徴としており、さらに、回路構成を複数の基本的論理回路から順に回路規模を縮小していくことにより、故障箇所を絞り込むことを特徴とする故障箇所の特定化方法である。

【0023】

【発明の実施の形態】前述のように、CMOS論理回路は回路内部に物理欠陥を有すると、一般的傾向として“Iddq (Quiescent Vdd Supply Current)”と称する論理の静止状態における電源電流に異常値が現れる。図1は物理故障の存在による貫通電流発生の様子を示す模式図であり、図中符号11はLSI、12はVdd、13はGND、14は物理故障箇所、15はテストベクタ、16はIddq (異常) である。すなわち、LSI 11の内部に物理故障箇所14が存在すると、任意のテストベクタ15により設定された論理により、その物理故障箇所14を介して、または、物理故障箇所14の影響をうけてVdd 12からGND 13への貫通電流即ち異常Iddq 16が発生する。

【0024】通常、ゲートアレイ品に代表されるASIC (Application Specific Integrated Circuits) の設計は、予め準備された“ブロック”と称する基本的な論理を構成する回路を組み合わせることにより所望の電気回路を実現す

る設計方法である。そのようなASICのCMOS回路内部に存在する故障個所の絞り込みは、テストベクタごとに変化する各ブロックの論理シミュレーション情報と、Iddq異常を発生させるテストベクタ番号とを用いることで可能となる。

【0025】説明はまず、各ブロックの論理シミュレーション情報について説明し、次に、Iddq異常を発生させるテストベクタ番号抽出方法について説明する。その後、本発明の手順を説明し、次に故障ブロック絞り込みのための方式について説明する。

【0026】1) LSIの各ブロックの論理シミュレーション情報(図2)

図2はLSIの各ブロックの論理シミュレーション結果をアレンジし直す過程の説明図であり、(a)はLSIの入出力状態を示す模式図、(b)はLSIの入出力論理関係を示すテストベクタ表、(c)はLSIの各ブロックを示す模式図、(d)はSIM(Systems Integration Model)上の時刻変化ごとのブロックの出力論理変化表、(e)はLSIの各ブロックからの出力を示す模式図、(f)は各ブロックごとのテストベクタに対する入力論理を表すダンプリストである。図中符号21はLSI、22は入力端子、23は出力端子、25はテストベクタ、26は出力、27はブロックB1、28はブロックB2、29はブロックBnである。

【0027】論理シミュレーションは、通常LSI21の入力端子22から入力するテストベクタ25に対応して出力端子23に出力する期待値を検証する論理検証のためのツールであり、通常、電気回路の動作確認と同時に、タイミングや内部遅延を検証するために用いられる。従って、必要となる検証データは各ブロックごとの出力端子に出力する論理と論理の時間変化情報のみで十分である。しかしながら、本手法に必要となるシミュレーションデータはテストベクタ番号ごとに変化する各ブロックごとの入力論理情報であり、通常の論理シミュレーション結果をアレンジし直さねばならない。

【0028】図2は論理シミュレーション結果を本手法で必要となる結果にアレンジし直す過程を解説した説明図であり、まず、各ブロックごとの出力端子に出力する論理と論理の時間変化情報とを、各ブロックごとの出力端子に出力する論理と論理のテストベクタ番号ごとの変化情報に直し、次に各ブロックごとの出力端子が次段のブロックの入力端子に接続される関係から、ブロックごとの入力端子に入力する論理情報と論理のテストベクタ番号ごとの変化情報に直した変遷を示している。

【0029】2) Iddq異常を発生させるテストベクタ番号(図3)

テストベクタごとのIddq異常の有無情報は故障品LSIの入力端子よりテストベクタを入力し、各テストベクタでの論理の静止状態におけるリーク電流値を測定

し、規格値を越えたリーク電流値をIddq異常値として識別したものが、Iddq異常有りとして抽出されるテストベクタ情報である。図3はテストベクタごとのIddq異常有無情報を示すグラフであり、図中符号31はIddqのしきい値、32はIddq(正常)、33はIddq(異常)である。X軸にテストベクタ番号(以降、TVno.と記す)をY軸にIddq値を示す。正常品のIddq値は規格値以下(例えば、正常状態において回路に貫通電流が発生しない時は1μA以下)であるのに対して、Iddq異常品は規格値の数倍から数千倍の異常な貫通電流が流れる。

【0030】3) 本発明の手順(図4)

図4は本発明の手順を示すフローチャートである。図中符号41はテストベクタ(TV)、42はLSIへの印加、43はIddq値の測定、44はIddq値異常のTV番号の抽出、45は回路接続情報、46は論理シミュレーション、47は内部回路ノード抽出のダンプ処理、48はTVごとの各ブロックの入力論理の抽出、49は演算処理、50は故障ブロックの抽出である。

【0031】本発明では、LSIの論理動作をテストするために準備されたテストベクタは、テストベクタごとに変化するLSIを構成する各ブロックごとの論理情報と、テストベクタごとのIddq異常の有無情報とを検出するために用いられる。

【0032】まず、テストベクタ41ごとに変化する各ブロックの入力論理情報48は上述した方式により、回路接続情報45からLSI回路検証のための論理シミュレーション46を用いて“ダンプ処理”47と称するLSIを構成する各ブロックのテキスト名を指定して出力する。そのテストベクタごとの各ブロックの入力論理情報48は上述したテストベクタごとのIddq異常の有無の情報44と組み合わせることで、各ブロックのテストベクタごとの演算処理49が実施され、故障個所を内蔵したブロックの抽出50が行なわれる。

【0033】そして、抽出された故障個所を内蔵したブロック50をもとに、そのブロックでのIddq異常の有無における論理情報からトランジスタレベルでの異常箇所を抽出し完了する。

【0034】4) 故障ブロック絞り込みのための方式

(図5)

図5は本発明による故障ブロックを絞り込むための基本的な考え方を述べる説明図であり、(a)はLSI内のブロックを示す模式図、(b)は各ブロックごとのダンプリスト、(c)は(b)のダンプリストに対応させたIddq異常有無情報を示すグラフである。図中符号51はLSI、52はブロックB1、53はブロックB2、54はブロックB3、55はブロックBn、56はテストベクタ、57はIddq(異常)、58はダンプリスト、59はIddq異常箇所である。

【0035】複数のブロック(B1、B2、B3、...

・・・、 B_n 、・・・)にて構成されるLSI51の入力端子よりテストベクタ56を入力する。その入力論理はそれらのブロックで論理を展開しながら出力端子に至る。

【0036】上述したダンプ処理により、各ブロックのテストベクタごとの論理状態が抽出される。この様子は図5(b)に示す各ブロックごとのダンプリスト58に示される。ところで、LSIにてテストベクタごとに検出されるIddq値は、テストベクタごとの各ブロック(B_1 、 B_2 、 B_3 、・・・、 B_n 、・・・)に発生するIddq値の合計であり、通常、規格値以内に収まっている。しかしながら、物理故障を内蔵しているブロックがそのブロック群に存在すると、そのブロックに発生する異常電流値がLSIでのIddq値異常として検出される。

【0037】図5(c)に示された上述のIddq値異常57が発生しているテストベクタ番号TVno. a、b、cは、各ブロックごとのダンプリストにおけるTVno. a、b、cにも対応しており、従って、各ブロックにてIddq異常表示されたテストベクタと正常状態でのテストベクタの入力論理を比較する(後述する)ことによりIddq異常を内蔵するブロックを抽出することができる。

【0038】5) 故障ブロックの抽出方式

次に、Iddq異常を内蔵するブロックを抽出する方法について説明する。ASICを構成するブロックは大きく2種類の回路に分類される。即ち組合わせ回路と順序回路である。組合わせ回路はそのブロックの入力端子に信号が印加されるとその論理が内部の回路を介して直接出力してくる回路形式であり、基本ゲート(AND、OR、NAND、インバータ回路等)からALUやADDER回路等の大きな規模の回路までである。また、順序回路はクロック信号に同期してデータが一旦回路内部に蓄えられ、次のクロック信号で出力するといった出力を行う回路形式であり、フリップ・フロップ、レジスタ回路やラッチ回路等がある。これら2種類の回路の故障診断は異なった方式で実施される。

【0039】5-1) 組合わせ回路に対する故障ブロックの抽出方式(図6)

まず、組合わせ回路に対する故障ブロックの抽出方式について図6にて説明する。図6は組合わせ回路の故障ブロックの抽出方法の説明図であり、(a)はダンプリストとして抽出された5入力端子を有する組合わせ回路のダンプリスト、(b)はIddq異常発生時のフローチャート、(c)はダンプリストとして抽出された5入力端子を有する組合わせ回路のダンプリスト、(d)はブロックを正常と判定するためのフローチャート、(e)はダンプリストとして抽出された5入力端子を有する組合わせ回路のダンプリスト、(f)はブロックに故障内蔵の疑いがあると判断するフローチャートである。

【0040】図に示すように、Iddq異常が発生する入力論理の組合わせは3種類(A、B、C)存在している。それらの種類に対するTV番号は、

入力論理の組合わせAは TV(a1)、TV(a2)、TV(a3)

入力論理の組合わせBは TV(b1)、TV(b2)

入力論理の組合わせCは TV(c1)、TV(c2)、TV(c3)

である。

【0041】これらの3種類の入力論理の組合わせと同じ入力論理の組合わせがIddq値が正常なTVに存在するかどうかコンピュータ(PCまたはEWS)を用いて調査する。

【0042】図6(c)、(d)はブロックを正常と判断するケースの1例である。入力論理の組合わせA、B、Cの全てが正常なTVに存在したとき、このブロックを正常と判断する。即ち、このブロックにおける“入力論理の組合わせA”と同一の入力論理の組合わせがIddq値が正常なTVに存在したとき、このブロックにおける“入力論理の組合わせA”は故障を示唆している入力論理ではないと判断する。次に、“入力論理の組合わせB”と同一の入力論理の組合わせがIddq値が正常なTVに存在したとき、上記と同様に“入力論理の組合わせB”は故障を示唆している入力論理ではないと判断する。以下、同様に“入力論理の組合わせC”と同一の入力論理の組合わせがIddq値が正常なTVに存在したとき上記と同様に“入力論理の組合わせC”は故障を示唆している入力論理ではないと判断し、全てのIddq値異常の入力論理の組合わせが故障を示唆していないので、このブロックは正常と判断してブロックの調査を終了する。

【0043】図6(e)、(f)はブロックに故障内蔵の疑いがあると判断するケースの1例である。上述と同様な調査を行い、入力論理の組合わせA、B、Cの内の少なくとも1組合わせ以上がIddq値が正常なTVにおける入力論理の組合わせに存在していない時に、このブロックに故障内蔵の疑いがあると判断する。

【0044】その理由について説明する。多重故障は1個のブロックに故障が全て内蔵しているか、複数のブロックに故障が内蔵しているかのどちらかである。

【0045】前者は、1個のブロックに故障が全て内蔵しているため、図6(e)、(f)における“ブロックに故障内蔵の疑いがあると判断するケース”を考えた時、これら3種類のIddq異常が発生する入力論理の組合わせは、それらの組合わせがあれば必ずIddq異常を発生し、従って、Iddq値が正常なTVにおける入力論理の組合わせには存在しないはずだからである。

【0046】後者は、複数のブロックに故障が内蔵しているケースであり、3種類の入力論理の組合わせ中の1種類(A)がIddq値の正常なTVにおける入力論理

の組合わせに存在する時、その入力論理の組合わせ

(A)は別のブロックに内蔵されたIddq異常の情報
が重なったものであると判断できる。そして、他の2種
類がIddq異常を発生するTVであり、故障を内蔵し
ている可能性ありと判断されるからである。

【0047】同様に、3種類の入力論理の組合わせ中の
2種類(A、B)がIddq値が正常なTVにおける入
力論理の組合わせに存在する時、それは別のブロックに
内蔵されたIddq異常の情報が重なったものであると
判断できる。そして、残った他の1種類はIddq異常
を発生するTVであり、故障を内蔵している可能性あり
と判断されるからである。

【0048】さらに、3種類の入力論理の組合わせ中の
全て(A、B、C)がIddq値が正常なTVにおける
入力論理の組合わせに存在する時、それは別のブロック
に内蔵されたIddq異常の情報が重なったものである
と判断できる。従ってこのブロックにはIddq異常を
発生するTVはなく、正常であると判断されるからであ
る。

【0049】5-2)順序回路に対する故障ブロックの
抽出方式(図7)

次に、順序回路に対する故障ブロックの抽出方式につ
いて図7にて説明する。図7は順序回路の故障ブロックの
抽出方法の説明図であり、(a)はダンブリストとして
抽出された5入力端子を有する順序回路のダンブリス
ト、(b)はIddq異常発生フローチャート、
(c)はダンブリストとして抽出された5入力端子を有
する順序回路のダンブリスト、(d)はブロックを正常
と判定するためのフローチャート、(e)はダンブリス
トとして抽出された5入力端子を有する順序回路のダン
ブリスト、(f)はブロックに故障内蔵の疑いがあると
判断するフローチャートである。

【0050】図7(a)、(b)に示すように、Idd
q異常が発生する入力論理の組合わせは7種類(A、
B、C、D、E、F、G)存在している。それらのTV
上での発生の様子には特徴があり、Iddq異常が発生
する入力論理の組合わせは連続したTVにてA、B、
C、Dの群とE、F、Gの群にて連続発生している。

【0051】このときは、これらの群(A、B、C、
D)と群(E、F、G)の固まりと同じ入力論理の組合
わせの固まりがIddq値が正常なTVに存在するかど
うか調査する。

【0052】図7(c)、(d)はブロックを正常と判
断するケースの1例である。入力論理の組合わせ群

(A、B、C、D)および(E、F、G)と同一の入力
論理の組合わせ群がIddq値が正常なTVに存在した
とき、このブロックを正常と判断する。即ち、入力論理
の組合わせ群(A、B、C、D)と同一の入力論理の組
合わせ群がIddq値が正常なTVに存在したとき、こ
のブロックにおける“入力論理の組合わせ群(A、B、

C、D)”は故障を示唆している入力論理ではないと判
断する。次に、入力論理の組合わせ群(E、F、G)と
同一の入力論理の組合わせ群がIddq値が正常なTV
に存在したとき、上記と同様に“入力論理の組合わせ群
(E、F、G)”は故障を示唆している入力論理ではな
いと判断し、全てのIddq値異常の入力論理の組合
わせ群が故障を示唆していないので、このブロックは正常
と判断してブロックの調査を終了する。

【0053】図7(e)、(f)はブロックに故障内蔵
の疑いがあると判断するケースの1例である。上述と同
様な調査を行い、入力論理の組合わせ群(A、B、C、
D)および、群(E、F、G)の内の少なくとも1組合
わせ群以上がIddq値が正常なTVにおける入力論理
の組合わせ群に存在していない時に、このブロックに故
障内蔵の疑いがあると判断する。

【0054】その理由については組合わせ回路での説明
と同様である。多重故障は1個のブロックに故障が全て
内蔵しているか、複数のブロックに故障が内蔵している
かのどちらかである。

【0055】前者は、1個のブロックに故障が全て内蔵
しているため、図7(e)、(f)における“ブロック
に故障内蔵の疑いがあると判断するケース”を考えた
時、これら2種類のIddq異常が発生する入力論理の
組合わせ群は、それらの組合わせ群があれば必ずIdd
q異常を発生し、従って、Iddq値が正常なTVにお
ける入力論理の組合わせ群には存在してないはずだから
である。

【0056】後者は、複数のブロックに故障が内蔵して
いるケースであり、2種類の入力論理の組合わせ中の1
種類群(A、B、C、D)がIddq値の正常なTVに
おける入力論理の組合わせ群に存在する時、その入力論
理の組合わせ群(A、B、C、D)は別のブロックに内
蔵されたIddq異常の情報が重なったものであると判
断できる。そして、他の群(E、F、G)はIddq異
常を発生する入力論理の組合わせ群であり、故障を内蔵
している可能性ありと判断されるからである。

【0057】以上、順序回路における診断方式を述べた
が、順序回路は連続したTVの他に、単一のTVにてI
ddq異常が発生しているケースがあり、例えば、Idd
q異常が発生する入力論理の組合わせは7種類(A、
B、C、D、E、F、G)存在している。Iddq異常
が発生する入力論理の組合わせは連続したTVでの
(A、B、C、D)の群と単一の入力論理の組合わせ
E、Fおよび、Gというケースである。この場合の調査
は連続したTVでの(A、B、C、D)の群、入力論理
の組合わせE、Fおよび、Gの4種類を考える。さら
に、連続したTVでの(A、B、C、D)の群に単一の
TVの入力論理の組合わせと同一の入力論理の組合
わせが存在しても(例えば、B=F)、これらは別個に考え
る。調査は、連続したTVでの(A、B、C、D)の群

にておこない、さらに単一のTV(F)での調査をおこなう。

【0058】この判断理由を説明するために、基本的な順序回路であるDタイプフリップ・フロップ（以降、DF/Fと記す）を用いて説明する。

【0059】図8は順序回路の診断方式の説明図であり、(a)はDタイプフリップ・フロップの回路図、(b)はDタイプフリップ・フロップの論理テーブルである。

【0060】図8(a)は、1個のインバータ回路、2個の2入力AND回路および、2個の2入力NOR回路から構成される2入力(D:Data端子、CLK:Clock端子)、2出力(Q、Q')の端子群を有するDF/Fの回路構成である(Q'=Qの反転論理を意味する)。図8(b)はその論理テーブル(真理値表)を示す。DF/Fにおいて、Q出力側2入力NOR回路が“H”出力になった時l d d q異常が発生したとすると、この異常は図8(b)に示す論理テーブル表のテストベクタTVno. 6、7、10、11、12、13においてl d d q異常が発生する。

【0061】これを単一の入力論理の組合わせとして見ると、l d d q異常の発生したTVの内、TVno. 7、11、13に注目したとき、同様な入力論理(D=0、CLK=0)がTVno. 3、5、9においても設定されている。しかしながら、これらTVno. 3、5、9の入力論理においてはl d d q異常は発生していない。その理由は順序回路はクロック信号に同期してデータが一旦回路内部に蓄えられ、次のクロック信号で出力するという回路形式のためである。

【0062】すなわち、図8(b)に示す論理テーブル表から明らかなように、テストベクタTVno. 7、11、13において保持されている論理はQ出力側2入力NOR回路が“H”および、Q'出力側2入力NOR回路が“L”出力となった論理であるのに対し、テストベクタTVno. 3、5、9において保持されている論理はQ出力側2入力NOR回路が“L”および、Q'出力側2入力NOR回路が“H”出力となった論理であるためである。従って、Q出力側2入力NOR回路が“H”出力になった時l d d q異常が発生する状態はテストベクタTVno. 7、11、13での保持状態で異常として検出されている。従って、順序回路における故障箇所検出の方式はDF/Fへの論理の設定と保持状態の入力論理を1つの組合わせ群(テストベクタTVno. 6、7)および{TVno. 10、11、12、13}として調査しなければならない。すなわち、順序回路の故障箇所検出の方式はl d d q異常の入力論理の組合わせ群が正常なテストベクタでの入力論理の組合わせ群の中に存在するかどうか調査しなければ、そのブロックに故障が存在するかどうかの判定がなされない。

【0063】6) 階層別分割による故障ブロック絞り込

み方式(図9)

図9はLSIを階層別に分割し、故障ブロックを絞り込む方式を示す説明図であり、図中91はLSI、92は順序回路、93、94、95は組合わせ回路、96はブロック(b1)、97はブロック(b2)、98はブロック(b3)、99は基本ブロックである。

【0064】階層別ブロック構成の場合は、設計時に使用されるライブラリと称する、予め準備されている基本的な回路構成をされたブロックの単位で解析を行う方式が一般的であるが、大規模化されたLSIにおいては膨大なブロック数となることが予測される。それに対応するために、全体を任意の大きさに再分割して、LSI内部のブロック構成を変える方式がある。その再構成で注意すべきことは、組合わせ回路と順序回路とを区別して1つの階層単位に階層分割しなければならないことである。理由は上述した組合わせ回路と順序回路での演算処理の方式が異なるためである。

【0065】図中、階層分割においては、組合わせ回路と順序回路が混在しているため、順序回路を境にして組合わせ回路を分割しやすい階層構成(a1、a2、a3)として故障箇所の絞り込みを行っている。次に、故障が発生しているとして抽出された階層ブロック(a1)93において、階層ブロック組を構成する階層構成b1、b2、b3にて故障箇所の絞り込みを行う。そして最後に最小単位である、“基本的論理回路構成”でのブロック(b2)97を抽出する。

【0066】7) トランジスタレベルの故障箇所の絞り込み方法(図10)

次に、“基本的論理回路構成”での基本ブロック99より、その回路を構成しているトランジスタレベルの故障箇所を絞り込む。基本的論理回路は一般に数トランジスタから数十トランジスタで構成される、OR、NOR、NAND、AND、インバータ回路、フリップフロップ回路といった基本的な論理動作を単位とする機能回路である。故障箇所の絞り込み方法は組合わせ回路、順序回路とも同じである。これらの基本的論理回路は入力論理に同期してどのトランジスタがON/OFFしているか明確に判定できるため、論理シミュレーションにより容易に故障トランジスタを抽出できる。以下に基本的論理回路の1つである、2入力NAND回路の例を用いて説明する。

【0067】図10は論理シミュレーションによる故障トランジスタの抽出の説明図であり、(a)は2入力NANDの回路図であり、(b)は2入力NAND回路の入力論理の全組合わせに注目した真理値表とl d d q値との相関表である。2入力にて構成する4つの入力論理の組合わせ(TV1、TV2、TV3、TV4)中、TV3(入力端子IN1にLowレベル、入力端子IN2にHighレベルが入力)にてl d d q異常が発生し、他の入力論理の組合わせ(TV1、TV2、TV4)は

正常であったとする。この情報を基に故障トランジスタを抽出する手法を説明する。通常、この故障トランジスタを抽出する方法は論理シミュレーションを用いて、各テストベクタにてどのトランジスタがON/OFFしているかを検証することにより実施する。

【0068】図10の2入力NAND回路図において、*

TVno.	PTr-1	PTr-2	NTr-1	NTr-2	Iddq値
TV1、	OFF	OFF	ON	ON	正常
TV2、	OFF	ON	ON	OFF	正常
TV3、	ON	OFF	OFF	ON	異常
TV4、	ON	ON	OFF	OFF	正常

と記述できる。

※入力NAND回路において2通りが考えられる。

【0069】さらに、Iddq異常が発生する通路は2※ 【0070】

- 1、 Vdd→ PTr1→ NTr1→ NTr2→ GND
- 2、 Vdd→ PTr2→ NTr1→ NTr2→ GND

これらの通路と上述のトランジスタのON/OFF状況との検証から、Iddq異常が発生する通路をトランジ★

★スタのON/OFF状況で書き表してみると、Iddq異常が発生する通路1は、

1、 Vdd→ PTr1→ NTr1→ NTr2→ GND	
TV1、	OFF ON ON 正常
TV2、	OFF ON OFF 正常
TV3、	ON <u>OFF</u> ON 異常
TV4、	ON <u>OFF</u> OFF 正常

となり、NTr1が異常となって常にON状態の時、貫通電流が発生する。

☆

☆【0071】また、Iddq異常が発生する通路2は、

2、 Vdd→ PTr2→ NTr1→ NTr2→ GND	
TV1、	OFF ON ON 正常
TV2、	ON ON OFF 正常
TV3、	<u>OFF</u> <u>OFF</u> ON 異常
TV4、	ON <u>OFF</u> OFF 正常

となり、PTr2およびNTr1が異常となって常にON状態となった時、貫通電流が発生する。しかしながら、TV1はIddq値は正常なため、PTr2に物理故障が内蔵していることと矛盾し、PTr2は正常と判断される。

【0072】以上、通路1、2の検証によりNTr1が異常となり常にON状態の時、貫通電流が発生することが判明する。

【0073】以上は4トランジスタ構成である2入力NAND回路についての例であるが、数十トランジスタにて構成された基本論理回路は上述した検証と同様の操作をコンピュータを用いた論理シミュレーションにより実施することにより短時間で異常トランジスタを抽出できる。

【0074】8) 組合わせ回路の故障箇所絞り込み方法(図11)

図11は組合わせ回路における故障箇所を絞り込む方法を示す説明図であり、(a)は真理値表とIddq値との相関表、(b)は(a)のIddq異常箇所の拡大表、(c)は故障素子絞り込みの模式図であり、図中符号111、112はテストベクタの変化、113は絞り

*テストベクタ(TV1、TV2、TV3、TV4)に対する、Pチャネルトランジスタ1、2(以降、PTr-1、PTr-2と記す)Nチャネルトランジスタ1、2(以降、NTr-1、NTr-2と記す)のON/OFF状況は

30 込んだ故障素子である。

【0075】Iddq異常が発生したテストベクタおよび、正常状態でのテストベクタを用いてIddq異常が発生したテストベクタにおける論理の変化を公知のシミュレーションにより抽出する方式である。特に重要となるデータはIddq異常が発生したテストベクタであり、組合わせ回路の場合、そのIddq異常が発生するテストベクタの前後において内部の論理の変化を用いて、公知の論理シミュレーションにより、故障内蔵素子を簡単に抽出できる。図11は前述した故障内蔵素子を簡単に抽出するための説明図であり、テストベクタTVno. (a-1)からTVno. (a)に変化{F(a-1, a)}した時、Iddq異常が発生する。この時のベクタの変化は入力ピンNO5のみが“0”から“1”へ変化し、他は変化していない。この論理の変化は論理シミュレーションにより変化素子を抽出できる。さらにテストベクタTVno. (a)からTVno. (a+1)に変化{F(a, a+1)}した時、正常状態に戻る。この時のベクタの変化は入力ピンNO8のみが“1”から“0”へ変化し、他は変化していない。前処理にて複数個の疑似故障素子が抽出されたとき、それ

らの疑似故障素子をさらに絞り込むことができる。その処理をIddq異常が発生した全テストベクタに関して実行することにより、故障内蔵素子を抽出することができる。

【0076】

【発明の効果】以上説明したように、本発明はIddq異常が発生したという現象を用いて、多重故障箇所を絞り込む方法であり、その方法は6つの大きな効果を有する。

【0077】第1は、出力端子異常の有無に関係なく故障箇所を絞り込めるという点である。これは本発明の一番大きな特徴である。

【0078】第2は、容易に故障箇所を絞り込むことができるという点である。すなわち、本方法の実行はLSI設計段階での検証ツールとして用いる論理シミュレーションを基にした各ブロックごとのダンプリストと、Iddq異常が発生したテストベクタ番号のみのデータでよいので、回路が解らなくても簡単に故障箇所を絞り込むことができる効果がある。

【0079】第3は、さらに、上述したデータは故障品のIddq異常が発生したテストベクタ番号のみでよいので、故障品がなくても解析が可能となる効果がある。

【0080】第4は、予め多重故障数が分からなくても、故障数および、その故障数に対応した故障箇所は演算処理にて特定化できるという点である。

【0081】第5は、高速に処理が可能である点である。本方法はコンピュータが得意とする演算処理のみでよいので、高速に処理が可能である。また、LSIが大規模になったとしても分割したブロック単位での演算が可能なので、コンピュータ容量の影響を受けないメリットがある。さらに、その演算はブロックごとにて独自に行うため、並列処理による短時間処理が可能である。

【0082】第6は、本発明の適用はIddq異常値の大きさに関係しないということである。解析に必要なデータはIddq異常が発生したテストベクタ番号であり、Iddq異常値の大きさに関係しないため、正常品とわずかな差の現れるサンプルにおいてもその発生箇所を絞り込むことが可能となる。

【図面の簡単な説明】

【図1】物理故障の存在による貫通電流発生の様子を示す模式図である。

【図2】LSIの各ブロックの論理シミュレーション結果をアレンジし直す過程の説明図である。(a)はLSIの入出力状態を示す模式図である。(b)はLSIの入出力論理関係を示すテストベクタ表である。(c)はLSIの各ブロックを示す模式図である。(d)はSIM(Systems Integration Model)上の時刻変化ごとのブロックの出力論理変化表である。(e)はLSIの各ブロックからの出力を示す模式図である。(f)は各ブロックごとのテストベクタに

対する入力論理を表すダンプリストである。

【図3】テストベクタごとのIddq異常有無情報を示すグラフである。

【図4】本発明の手順を示すフローチャートである。

【図5】本発明による故障ブロックを絞り込むための基本的な考え方を述べる説明図である。(a)はLSI内のブロックを示す模式図である。(b)は各ブロックごとのテストベクタ表である。(c)は(b)のテストベクタ表に対応させたIddq異常有無情報を示すグラフである。

【図6】組み合わせ回路の故障ブロックの抽出方法の説明図である。(a)はダンプリストとして抽出された5入力端子を有する組み合わせ回路のダンプリストである。

(b)はIddq異常発生時のフローチャートである。

(c)はダンプリストとして抽出された5入力端子を有する組み合わせ回路のダンプリストである。(d)はブロックを正常と判定するためのフローチャートである。

(e)はダンプリストとして抽出された5入力端子を有する組み合わせ回路のダンプリストである。(f)はブロックに故障内蔵の疑いがあると判断するフローチャートである。

【図7】順序回路の故障ブロックの抽出方法の説明図である。(a)はダンプリストとして抽出された5入力端子を有する順序回路のダンプリストである。(b)はIddq異常発生時のフローチャートである。(c)はダンプリストとして抽出された5入力端子を有する順序回路のダンプリストである。(d)はブロックを正常と判定するためのフローチャートである。(e)はダンプリストとして抽出された5入力端子を有する順序回路のダンプリストである。(f)はブロックに故障内蔵の疑いがあると判断するフローチャートである。

【図8】順序回路の診断方式の説明図である。(a)はDタイプフリップ・フロップの回路図である。(b)はDタイプフリップ・フロップの論理テーブルである。

【図9】LSIを階層別に分割し、故障ブロックを絞り込む方式を示す説明図である。

【図10】論理シミュレーションによる故障トランジスタの抽出の説明図である。(a)は2入力NANDの回路図である。(b)は2入力NAND回路の入力論理の全組み合わせに注目した真理値表とIddq値との相関表である。

【図11】組み合わせ回路における故障箇所を絞り込む方法を示す説明図である。(a)は真理値表とIddq値との相関表である。(b)は(a)のIddq異常箇所の拡大表である。(c)は故障素子絞り込みの模式図である。

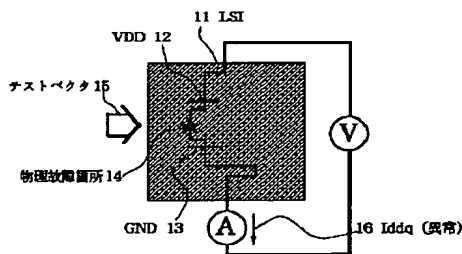
【図12】従来例の故障辞書作成による故障シミュレーション方法の流れブロック図である。

【符号の説明】

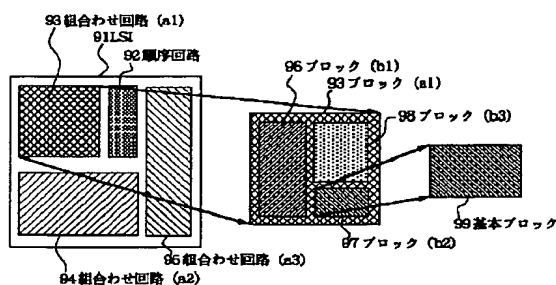
11、21、51、91 LSI

- 12 Vdd
- 13 GND
- 14 物理故障箇所
- 15、25 テストベクタ
- 16 Iddq (異常)
- 22 入力端子
- 23 出力端子
- 26 出力
- 27 ブロックB1
- 28 ブロックB2
- 29 ブロックBn
- 31 Iddqのしきい値
- 32 Iddq (正常)
- 33 Iddq (異常)
- 41 テストベクタ (TV)
- 42 LSIへの印加
- 43 Iddq値の測定
- 44 Iddq値異常のTV番号の抽出
- 45 回路接続情報
- 46 論理シミュレーション
- 47 内部回路ノード抽出のダンプ処理
- 48 TVごとの各ブロックの入力論理の抽出
- 49 演算処理

【図1】



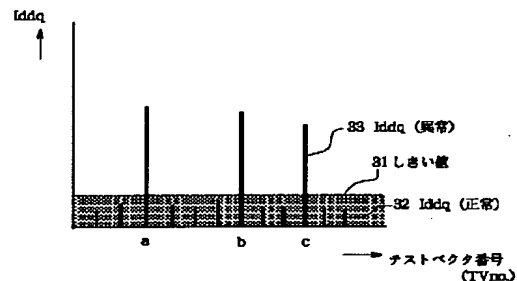
【図9】



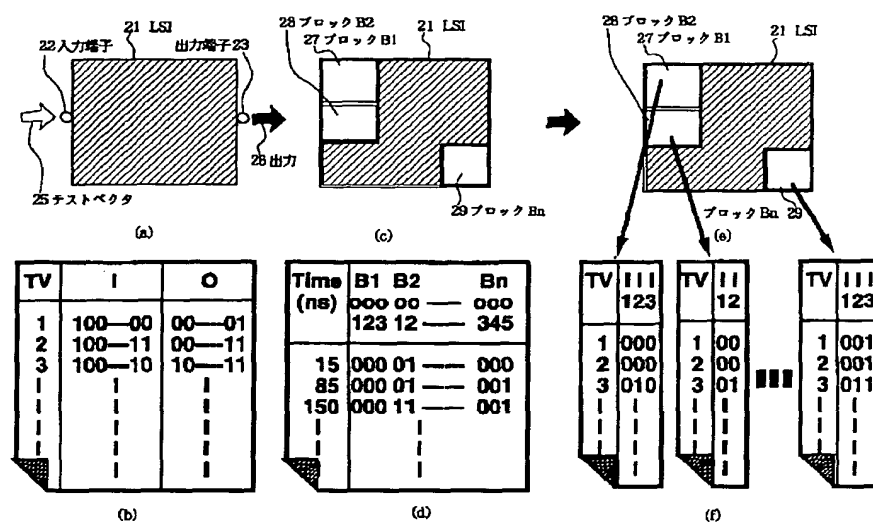
- * 50 故障ブロックの抽出
- 52 ブロックB1
- 53 ブロックB2
- 54 ブロックB3
- 55 ブロックBn
- 56 テストベクタ
- 57 Iddq (異常)
- 58 ダンプリスト
- 59 Iddq異常箇所
- 10 92 順序回路
- 93、94、95 組合わせ回路
- 96 ブロック (b1)
- 97 ブロック (b2)
- 98 ブロック (b3)
- 99 基本ブロック
- 111、112 テストベクタの変化
- 113 絞り込んだ故障素子
- 121 テストベクタ
- 122 故障定義
- 20 123、124 LSIの論理接続
- 125、126 論理シミュレーション
- 127 比較

*

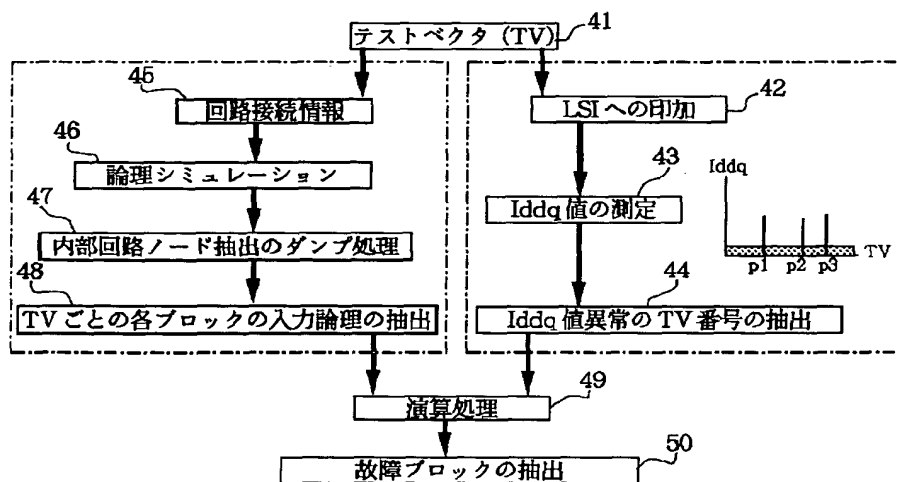
【図3】



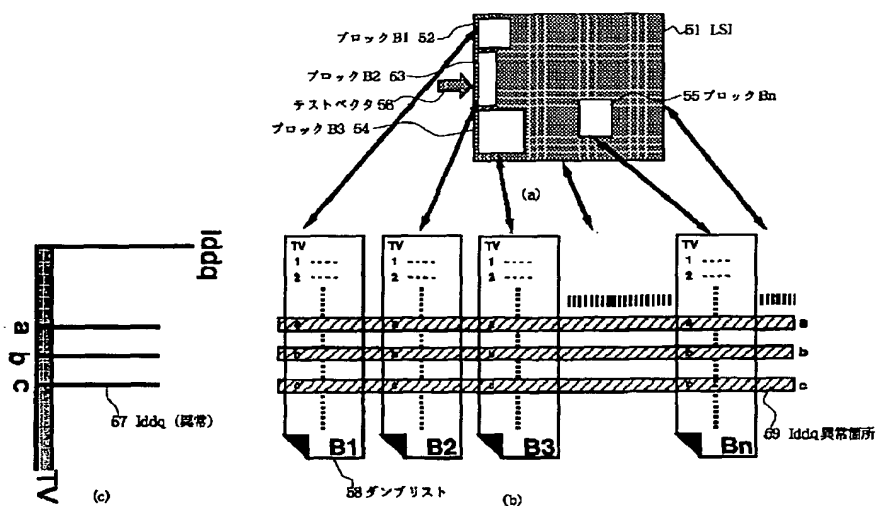
【図2】



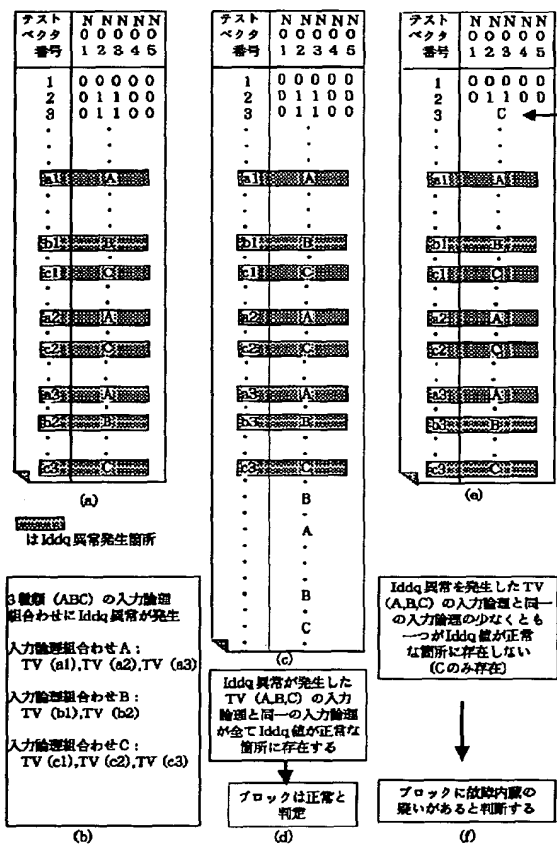
【図4】



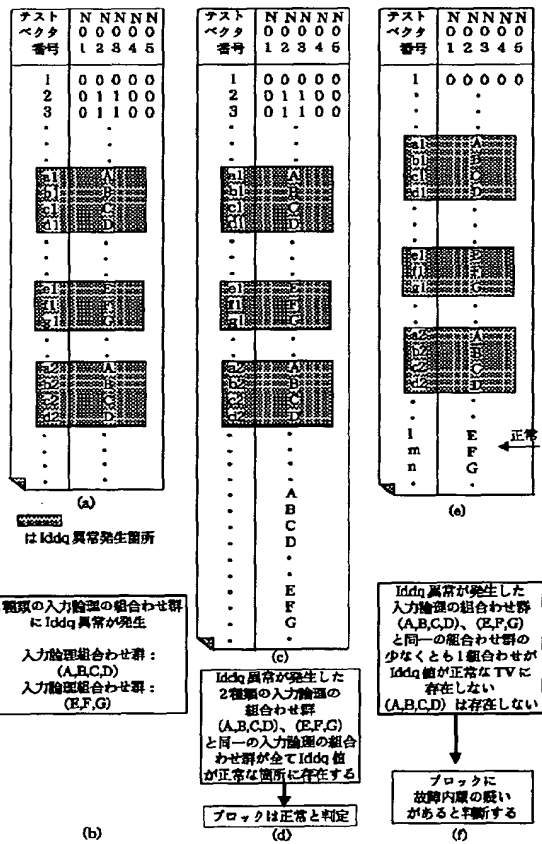
【図5】



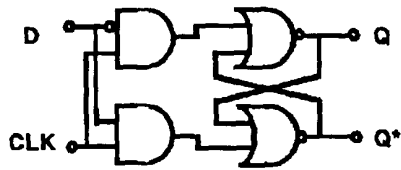
【図6】



【圖 7】



【図8】

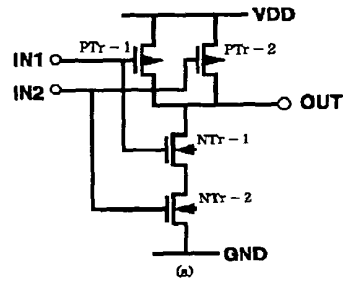


(a)

TVno.	D CLK	Q Q*
1	0 0	Q Q*
2	0 1	0 1
3	0 0	0 1
4	0 1	0 1
5	0 0	0 1
6	0 1	0 1
7	0 0	0 1
8	0 1	0 1
9	0 0	0 1

(b)

【図10】



(a)

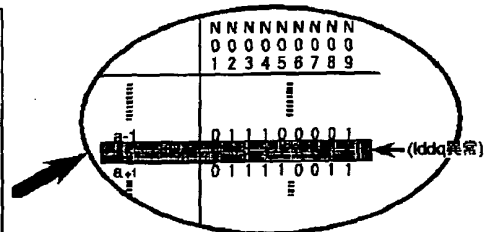
TVno.	IN1	IN2	OUT
1	H	H	L
2	H	L	H
3	L	H	H
4	L	L	H

(b)

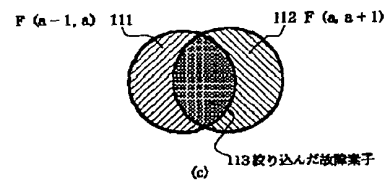
【図11】

テストベクタ 番号	NNNNNNNNN 000000000 123456789
1	000001000
2	011001000
3	011001001
4	011001001
5	011001001
6	011001001
7	011001001
8	011001001
9	011001001

(a)



(b)



(c)

【図12】

